

500.43408X00

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant(s): YAMADA, et al

Serial No.:

Filed: January 20, 2004

Title: MODULAR COMPUTER SYSTEM AND I/O MODULE

Group:

LETTER CLAIMING RIGHT OF PRIORITY

Mail Stop Patent Application
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

January 20, 2004

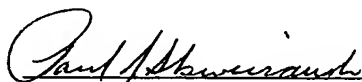
Sir:

Under the provisions of 35 USC 119 and 37 CFR 1.55, the applicant(s) hereby claim(s) the right of priority based on Japanese Patent Application No.(s) 2003-118714 filed April 23, 2003.

A certified copy of said Japanese Application is attached.

Respectfully submitted,

ANTONELLI, TERRY, STOUT & KRAUS, LLP



Paul J. Skwierawski
Registration No. 32,173

PJS/nac
Attachment
(703) 312-6600

日本国特許庁
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出願年月日 2003年 4月23日
Date of Application:

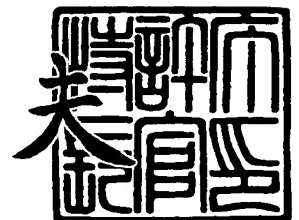
出願番号 特願2003-118714
Application Number:
[ST. 10/C]: [JP2003-118714]

出願人 株式会社日立製作所
Applicant(s):

2003年11月26日

特許庁長官
Commissioner,
Japan Patent Office

今井 康



【書類名】 特許願

【整理番号】 PE28989

【提出日】 平成15年 4月23日

【あて先】 特許庁長官 殿

【国際特許分類】 H01R 23/68

【発明者】

【住所又は居所】 茨城県日立市大みか町七丁目 1 番 1 号
株式会社日立製作所 日立研究所内

【氏名】 山田 勉

【発明者】

【住所又は居所】 茨城県日立市大みか町七丁目 1 番 1 号
株式会社日立製作所 日立研究所内

【氏名】 中三川 哲明

【発明者】

【住所又は居所】 茨城県日立市大みか町七丁目 1 番 1 号
株式会社日立製作所 日立研究所内

【氏名】 遠藤 浩通

【発明者】

【住所又は居所】 茨城県日立市大みか町七丁目 1 番 1 号
株式会社日立製作所 日立研究所内

【氏名】 松本 典剛

【発明者】

【住所又は居所】 茨城県日立市大みか町五丁目 2 番 1 号
株式会社日立製作所 情報制御システム事業部内

【氏名】 笠嶋 広和

【特許出願人】

【識別番号】 000005108

【氏名又は名称】 株式会社日立製作所

【代理人】

【識別番号】 100098017

【弁理士】

【氏名又は名称】 吉岡 宏嗣

【手数料の表示】

【予納台帳番号】 055181

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 モジュール型計算機システム及び I / O モジュール

【特許請求の範囲】

【請求項 1】 プロセッサが搭載された演算処理モジュールと複数の I / O モジュールとをバスを形成するコネクタを介して層状に接続して構成されるモジュール型計算機システムにおいて、前記各 I / O モジュールは、演算処理モジュール側コネクタの同一位置の端子から入力されるモジュール選択信号を有効とするモジュール排他選択部と、該モジュール排他選択部から出力されるモジュール選択有効信号に基づいて自己の I / O モジュールの識別情報を前記コネクタの所定端子に出力する識別情報出力部とを備えてなることを特徴とするモジュール型計算機システム。

【請求項 2】 前記演算処理モジュールは、前記 I / O モジュールが接続されるコネクタ端子に前記モジュール選択信号を出力するモジュール選択信号出力部と、前記コネクタの所定端子に出力される前記識別情報を取込む識別情報入力部とを備え、前記モジュール選択信号出力部は前記演算処理モジュールに接続された複数の前記 I / O モジュールに前記モジュール選択信号を順次出力するものであり、前記識別情報入力部は前記モジュール選択信号の出力順に従って前記 I / O モジュールと前記識別情報に対応付けて認識することを特徴とする請求項 1 に記載のモジュール型計算機システム。

【請求項 3】 前記演算処理モジュールは、前記識別情報入力部により認識された前記 I / O モジュールと前記識別情報の対応に従って、予め設定されたバス制御パラメータと前記 I / O モジュールのデバイスドライバとをメモリから読み出して、前記 I / O モジュールにアクセスすることを特徴とする請求項 2 に記載のモジュール型計算機システム。

【請求項 4】 プロセッサが搭載された演算処理モジュールと複数の I / O モジュールとをバスを形成するコネクタを介して層状に接続して構成されるモジュール型計算機システムにおいて、前記各 I / O モジュールは、演算処理モジュール側コネクタから入力されるモジュール選択信号が自己のモジュールを選択する信号か否か判別し、自己のモジュールを選択する信号のときは前記モジュール

選択信号の有効信号を出力し、演算処理モジュール側コネクタから入力されるモジュール選択信号が他のモジュールを選択する信号のときは、当該モジュール選択信号を反演算処理モジュール側コネクタの端子のうち、前記自己のモジュールを選択する信号が入力された前記演算処理モジュール側コネクタの端子と同一位置の端子に伝送するモジュール排他選択部と、前記有効信号に基づいて自己のモジュールの識別情報を前記コネクタの所定端子に出力する識別情報出力部とを有することを特徴とするモジュール型計算機システム。

【請求項 5】 前記モジュール排他選択部は、演算処理モジュール側の複数のコネクタ端子に接続された複数の配線を有し、該配線の 1 つを前記識別情報出力部に接続し、他の配線の 1 つを反演算処理モジュール側の複数のコネクタ端子のうち、前記自己のモジュールを選択するモジュール選択信号が入力された前記コネクタ端子と同一位置のコネクタ端子に接続して構成されることを特徴とする請求項 4 に記載のモジュール型計算機システム。

【請求項 6】 前記モジュール排他選択部は、D 型フリップフロップの D 端子を演算処理モジュール側のコネクタ端子の 1 つに接続し、前記 D 型フリップフロップの Q 端子を前記識別情報出力部と反演算処理モジュール側のコネクタ端子のうち、前記 D 端子が接続された前記コネクタ端子と同一位置の端子に接続し、クロック端子を演算処理モジュール側と反演算処理モジュール側とのコネクタ端子が共通接続される端子に接続して構成されることを特徴とする請求項 4 に記載のモジュール型計算機システム。

【請求項 7】 前記演算処理モジュールは、前記クロック端子が接続された前記コネクタ端子をクロック信号により駆動するとともに、前記 D 端子が接続されたコネクタ端子をイネーブル信号により駆動することを特徴とする請求項 6 に記載のモジュール型計算機システム。

【請求項 8】 前記識別情報出力部は、自己のモジュールの識別情報を生成する識別情報生成部と、該識別情報生成部で生成された前記識別情報を前記有効信号に基づいて前記コネクタの所定端子に出力する出力活性部とを有して構成されることを特徴とする請求項 4 に記載のモジュール型計算機システム。

【請求項 9】 前記出力活性部は、前記識別情報を入力とし、前記有効信号

により開かれるゲート素子を有して構成されることを特徴とする請求項 8 に記載のモジュール型計算機システム。

【請求項 10】 前記識別情報生成部は、前記有効信号とクロック信号に基づいてシリアル信号の前記識別情報を生成することを特徴とする請求項 8 に記載のモジュール型計算機システム。

【請求項 11】 前記識別情報出力部は、前記有効信号により駆動される配線と前記コネクタの複数の所定端子とを前記識別情報に応じて P N 接合素子を介して接続して構成されることを特徴とする請求項 4 に記載のモジュール型計算機システム。

【請求項 12】 前記演算処理モジュールは、前記 I / O モジュールが接続されるコネクタ端子に前記モジュール選択信号を出力するモジュール選択信号出力部と、前記コネクタの所定端子に出力される前記識別情報を取込む識別情報入力部とを備え、前記モジュール選択信号出力部は前記演算処理モジュールに接続された複数の前記 I / O モジュールに前記モジュール選択信号を順次出力するものであり、前記識別情報入力部は前記モジュール選択信号の出力順に従って前記 I / O モジュールと前記識別情報を対応付けて認識することを特徴とする請求項 4 に記載のモジュール型計算機システム。

【請求項 13】 前記演算処理モジュールは、前記識別情報入力部により認識された前記 I / O モジュールと前記識別情報の対応に従って、予め設定されたバス制御パラメータと前記 I / O モジュールのデバイスドライバとをメモリから読み出して、前記 I / O モジュールにアクセスすることを特徴とする請求項 12 に記載のモジュール型計算機システム。

【請求項 14】 演算処理モジュールにバスを形成するコネクタを介して層状に接続される I / O モジュールにおいて、演算処理モジュール側コネクタの同一位置の端子から入力されるモジュール選択信号を有効とするモジュール排他選択部と、該モジュール排他選択部から出力されるモジュール選択有効信号に基づいて自己の I / O モジュールの識別情報を前記コネクタの所定端子に出力する識別情報出力部とを備えてなることを特徴とする I / O モジュール。

【請求項 15】 前記モジュール排他選択部は、一方側のコネクタに接続さ

れた複数の配線を有し、該配線の 1 つを前記識別情報出力部に接続し、他の配線の 1 つを他方側のコネクタの複数の端子のうち、前記識別情報出力部に接続した配線が接続された前記コネクタの端子と同一位置の端子に接続して構成されることを特徴とする請求項 14 に記載の I/O モジュール。

【請求項 16】 前記モジュール排他選択部は、D 型フリップフロップの D 端子を演算処理モジュール側のコネクタ端子の 1 つに接続し、前記 D 型フリップフロップの Q 端子を前記識別情報出力部と反演算処理モジュール側のコネクタ端子のうち、前記 D 端子が接続された前記コネクタ端子と同一位置の端子に接続し、クロック端子を演算処理モジュール側と反演算処理モジュール側とのコネクタ端子が共通接続される端子に接続して構成されることを特徴とする請求項 14 に記載の I/O モジュール。

【請求項 17】 前記演算処理モジュールは、前記クロック端子が接続された前記コネクタ端子をクロック信号により駆動するとともに、前記 D 端子が接続されたコネクタ端子をイネーブル信号により駆動することを特徴とする請求項 16 に記載の I/O モジュール。

【請求項 18】 前記識別情報出力部は、自己のモジュールの識別情報を生成する識別情報生成部と、該識別情報生成部で生成された前記識別情報を前記有効信号に基づいて前記コネクタの所定端子に出力する出力活性部とを有して構成されることを特徴とする請求項 14 に記載の I/O モジュール。

【請求項 19】 前記出力活性部は、前記識別情報を入力とし、前記有効信号により開かれるゲート素子を有して構成されることを特徴とする請求項 18 に記載の I/O モジュール。

【請求項 20】 前記識別情報出力部は、前記有効信号により駆動される配線と前記コネクタの複数の所定端子とを前記識別情報に応じて P-N 接合素子を介して接続して構成されることを特徴とする請求項 14 に記載の I/O モジュール。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、複数の機能モジュールを組み合わせて構築するモジュール型計算機システムに係り、より詳細には、産業用機器やロボット等に組み込まれるコントローラに好適なモジュール型計算機システムに関する。

【0002】

【従来の技術】

複数の機能モジュールを組み合わせて計算機システムを構築する方式が種々提案されている（例えば、非特許文献1）。この文献1によれば、例えば、プロセッサを搭載した演算処理モジュールと、この演算処理モジュールと周辺機器とのインターフェースデバイスを搭載したI/Oモジュールの基板を、バスを形成するコネクタを介して積層して構成している。このようなモジュール方式の計算機システムにおいて、演算処理モジュールに搭載されたプロセッサは、I/Oモジュール等のシステムデバイスを管理するため、例えばI/Oモジュールに搭載されたI/Oデバイスの種類を識別し、オペレーティングシステム（OS）がI/Oデバイスを駆動するために利用するデバイスドライバを特定する必要がある。つまり、オペレーティングシステム（OS）は、I/Oデバイスの識別コードに対応させてデバイスドライバが設定されたテーブルを検索し、識別されたI/Oドライバのデバイスドライバを特定してメモリ上に展開することにより、I/Oデバイスを初期化したり、要求に応じてI/Oデバイスにアクセス処理をする。

【0003】

上記文献1においては、プロセッサが接続されたI/Oデバイスを識別するため、IDSEL信号を用いてI/Oモジュールを選択するとともに、I/Oモジュールに含まれるPCIデバイスに設けられたコンフィギュレーションレジスタの値を読み取ってPCIデバイスを認識するようにしている。この場合、プロセッサ側のPICデバイスとI/O側のPICデバイスとをPICバスにより接続し、IDSEL信号により対象のPICデバイスを物理的に指定するため、PCIバス仕様に準拠するコネクタをモジュール基板上に配置し、プロセッサ側のPICデバイスとI/O側のPICデバイスとの間で、IDSEL信号を個別に配線することになる。このようなPCI方式のモジュールは、I/Oデバイスを適切に初期化する機能であるプラグアンドプレイ機能を有し、かつ高機能であり、

しかも柔軟な設定ができる利点がある。このようなモジュール構成でプラグアンドプレイ機能を有する従来の方式として、ISAバス方式のデバイス認識手法等が知られている。

【0004】

【非特許文献1】

PC/104-plus Specification Version1.2:PC/104 Embedded Consortium制定（2001年8月）

【発明が解決しようとする課題】

ところが、文献1に記載される従来のモジュール化手法は、産業用機器やロボット等に組み込まれるコントローラ（以下、組込みコントローラという。）に用いる計算機システムのように、任意のI/Oモジュールが組み込まれることについては配慮されていない。すなわち、従来のモジュール化技術は、具体的な用途を特定した専用計算機システムのように、組み込まれるI/Oデバイス等が特定されている場合は、設計時間や手間などはあまり問題とならないが、任意のI/Oモジュールが組み込まれる場合は、設計に時間がかかったり、製造等に手間がかかるという問題がある。

【0005】

すなわち、産業用機器やロボット等に組み込まれる組込みコントローラに用いる計算機システムの場合は、量産時のコストを最小とするために、一般に、必要最小限の部品のみで構成する要請が強い。しかも、コントローラが組み込まれる産業用機器の小型化の要請から、実装容積を極力小さくしなければならないため、組み込むモジュールを可能な限り小型化できることが望まれる。これらのことから、組込みコントローラに用いる場合は、各モジュールをできるだけ単機能とし、必要に応じて種々の機能を有する複数のI/Oモジュール等を組み合わせて、所望の機能を備えた計算機システムを構築するのが望ましい。

【0006】

しかしながら、従来のPCI方式の場合は、PCIバスの信号線数に応じてコネクタのピンが多くなり、かつ各I/OモジュールにPCIバス制御を行うインテリジェント素子を設けなければならないから、基板の実装面積が増大して、小

型化が制限されるという問題がある。また、各モジュールを接続するバスは、P C Iバス等の特定された単一のバス方式を前提とするため、バス方式が違う複数のデバイスを同一バス上で利用すると、デバイスを認識してプラグアンドプレイ機能を実現することができない。

【0 0 0 7】

したがって、組込みコントローラのように任意の I / Oモジュール等が組み込まれることを想定し、かつプラグアンドプレイ機能を持たせるには、特定のバス方式に限定されずに組み込まれた I / Oモジュールのデバイスを識別する手法が望まれる。

【0 0 0 8】

また、従来のモジュール化手法によれば、プロセッサ側の P I Cデバイスと I / O側の P I Cデバイスとの間で、 I D S E L信号を個別に配線する必要があるため、バス上に複数の I D S E L信号線を設け、モジュールに接続する I D S E L信号線をモジュール毎に個別に設定することになる。この設定は、一般に、手作業によりモジュール毎に行わなければならないため、作業工数や検査工数が増える。また、スイッチによる実装面積や価格が増大する問題があり、かつ誤設定を完全になくすことは困難である。

【0 0 0 9】

そこで、本発明は、特定のバス方式に制約されることなく、計算機システムに組み込まれた I / Oモジュールの種類等を識別することを課題とする。

【0 0 1 0】

【課題を解決するための手段】

本発明は、上記課題を解決するため、演算処理モジュールにコネクタを介して層状に接続される各 I / Oモジュールにモジュール排他選択部を設け、このモジュール排他選択部により演算処理モジュール側コネクタの同一位置の端子から入力されるモジュール選択信号のみを有効と判別するようにしたことを特徴とする。そして、演算処理モジュールから順次出力されるモジュール選択信号を、 I / Oモジュールの接続順に応じて演算処理モジュール側コネクタの同一位置の端子に入力することにより、特定のバス方式に制約されることなく、簡単なモジュール

ル選択信号及び簡単な接続構成ないし回路構成によって唯一の各 I/O モジュールを選択できる。また、モジュール排他選択部によって有効と判別された場合に出力されるモジュール選択有効信号に基づいて、自己の I/O モジュールの識別情報をコネクタの所定端子に出力する識別情報出力部を設けることにより、演算処理モジュールは特定のバス方式に制約されることなく、前記コネクタの所定端子に対応する端子に接続された識別情報入力部により I/O モジュールの識別情報を獲得できる。例えば、接続順に従って出力するモジュール選択信号に対応した I/O モジュールの識別情報を獲得できる。

【0011】

これにより、演算処理モジュールは、獲得した I/O モジュールと識別情報の対応関係に従って、予め設定されたバス制御パラメータと I/O モジュールのデバイスドライバとをメモリから読み出して I/O モジュールにアクセスすることができる。また、本発明によれば、各 I/O モジュールの識別に係るモジュール排他選択部を同一の構成にできるから、モジュール毎に手作業による設定が不要で、同一回路のモジュールを組み合わせた場合でもデバイスを識別することができる。

【0012】

【発明の実施の形態】

以下、本発明の実施の形態について図面を用いて説明する。

（第1実施の形態）

図1に、本発明のモジュール型計算機システムの第1実施形態の全体構成を示す。本実施形態の計算機システムは、演算処理モジュール1と、複数の I/O モジュール2（図では、 n 個の I/O モジュール $2-1 \sim 2-n$ ）を有して構成される。演算処理モジュール1は、計算機システムの構成を管理するものであり、モジュール選択信号出力部10と、モジュール ID 入力部20と、設定パラメータテーブル50と、設定処理部60と、通信制御部70とを有して構成されている。

【0013】

モジュール選択信号出力部10は、I/O モジュール $2-1 \sim 2-n$ のうち任

意の 1 つを選択するためにモジュール選択信号 30 をコネクタの信号線 110 に出力する。モジュール I D 入力部 20 は、I/O モジュール 2 からモジュール I D バス 400 に出力されるモジュール I D 信号 40 を取り込む。なお、モジュール選択信号出力部 10 は、モジュール選択信号 30 を駆動するための複数の出力バッファを有する。また、モジュール I D 入力部 20 は、モジュール I D 信号 40 の状態を取り込むために複数の入力バッファを有する。これらモジュール選択信号出力部 10 とモジュール I D 入力部 20 とは、具体的にはプロセッサとメモリ、複数の I/O 接点入出力ピンにより構成される。モジュール選択信号出力部 10 とモジュール I D 入力部 20 とは、それぞれ個別、あるいは同一のプロセッサとメモリで制御される。プロセッサは、メモリに格納される命令列に従い複数の I/O 接点入出力ピンを駆動し、あるいは読み取る。

【0014】

I/O モジュール 2 は、一般に、演算処理モジュール 1 から入出力要求を受け付けて、演算・保持・計算機外部への入出力を実行する。I/O モジュール 2 は、モジュール排他選択部 100 と、出力活性部 200 及び I D 生成部 300 とからなる I D 情報出力部と、I/O デバイスである回路装置 50 とを有して構成されている。

【0015】

モジュール排他選択部 100 は、隣接する他のモジュールからモジュール選択信号 30 を入力し、そのモジュール選択信号 30 が自己のモジュールに対する選択信号か否かを判別し、その結果をモジュール選択有効信号 130 に反映する。例えば、自己の I/O モジュールが選択された場合、モジュール排他選択部 100 は、モジュール選択有効信号 130 の電位を有効状態を表す高電位（High レベル）にする。本明細書では、信号線の電位を有効状態に駆動することを「アサート」と称し、無効状態に駆動することを「ネゲート」と称する。I/O モジュール 2-1 のモジュール排他選択部 100 は、他方に隣接する I/O モジュール 2-2 にモジュール選択信号線の信号線 120 を介してモジュール選択信号 30 を出力する。本発明は、一の I/O モジュールの信号線 110 に入力されるモジュール選択信号 30 と、一の I/O モジュールの信号線 120 を介して他の I/O

モジュールに入力されるモジュール選択信号 30 とを異なる状態の信号とすることにより特徴がある。この点については、後述する。

【0016】

ID 生成部 300 は、I/O モジュール 2 を特定するための識別コードである ID を生成して、ID データ信号 140 を出力する。出力活性部 200 は、モジュール選択有効信号 130 を受けて、その信号が有効状態の場合は ID データ信号 140 をモジュール ID バス 400 に出力する。一方、モジュール選択有効信号 130 が無効状態の場合はモジュール ID バス 400 に ID データ信号 140 を出力しない。これにより、モジュール ID バス 400 上には唯一選択された I/O モジュール 2 の ID データ信号 140 のみが出力される。

【0017】

このように構成されることから、演算処理モジュール 1 は、I/O モジュール 2-1 ~ 2-n のうち、任意の I/O モジュールを選択するモジュール選択信号 30 を出力し、この信号により選択された I/O モジュールから ID データ信号 140 をモジュール ID バス 400 に出力させることにより、計算機システムに属する全ての I/O モジュール 2 の ID 情報を対応付けて獲得することが可能となる。また、I/O モジュール 2 は、モジュール選択信号 30 が入力される信号線 110 と、モジュール選択信号 30 を出力する信号線 120 と、モジュール ID バス 400 を、コネクタを介して隣接する I/O モジュール 2 と接続しあうことにより、信号線の増大を抑えることができる。

【0018】

ここで、図 1 のモジュール排他選択部 100 の具体的な一実施例を図 2 を用いて説明する。本発明は、モジュール排他選択部 100 を同一回路で構成することにより特徴とする。つまり、モジュール排他選択部 100 は、演算処理モジュール側コネクタの同一位置の端子に接続された信号線 110 からモジュール選択信号 30 が入力されるように構成されている。また、I/O モジュール 2 が互いにコネクタを介して層状に連結されていても、演算処理モジュール 1 より任意の位置に接続された I/O モジュール 2 を選択ができることが、モジュール排他選択部 100 の特徴である。

【0019】

図2の実施例は、モジュール排他選択部100を配線方法によって実現している。演算モジュール側に隣接するI/Oモジュール2から信号線110を介してモジュール選択信号30が入力される。そのうちの信号線110-1をモジュール選択有効信号130とする。残りの信号線110-x（ただし $x=2\sim n$ ）は、1本ずつ信号線をずらして反演算モジュール側に隣接するI/Oモジュール2の信号線120- $\{x-1\}$ に接続する。信号線120-nは、開放しても、信号線110-1と接続してもよい。すなわち、各I/Oモジュール2のモジュール排他選択部100は、演算処理モジュール側コネクタの同一位置の端子（信号線110-1）からモジュール選択信号30が入力されたときにモジュール選択有効信号130を有効状態にするように構成されている。したがって、本実施例のモジュール排他選択部100の場合は、演算処理モジュール1は信号線110-1 $\sim n$ に対応させて、モジュール選択信号30-1 $\sim n$ を順次出力することにより、I/Oモジュール2-1 $\sim n$ を順次選択できる。例えば、演算処理モジュールに最も近いI/Oモジュール2-1を選択するために、モジュール選択信号出力部10はモジュール選択信号30-1をHighレベルに駆動し、他のモジュール選択信号30-2 $\sim n$ をLowレベルとする。これにより、I/Oモジュール2-1のモジュール選択有効信号130はモジュール排他選択部100によりアサートされ、他のI/Oモジュール2-2 $\sim n$ のモジュール選択有効信号130はモジュール排他選択部100によりネゲートされる。同様に、演算処理モジュール1からx番目に離れたモジュールを選択するには、モジュール選択信号出力部10はモジュール選択信号30-xのみをHighに駆動すればよい。

【0020】

次に、モジュール排他選択部100によりアサートされモジュール選択有効信号130により、I/Oモジュール2からIDデータ信号140をモジュールIDバス400に出力させる識別情報出力部の実施例を図3、図4を用いて説明する。図3は、ID生成部300の構成例を示し、図4は、出力活性部200の構成例を示す。図3に示すように、ID生成部300は、I/Oモジュール2の種類や機能を特定する識別情報のIDデータを生成するものであり、ID生成子3

10-1～310-mを備え、各ID生成子からIDデータ信号140-1～140-mが出力される。ID生成子310は、出力するレベルを個別に設定可能な回路であり、出力するレベルとしてはHighレベルあるいはLowレベルの2値、あるいは任意の電圧レベル（多値）であってもよい。多値の電圧レベルの場合は、ID情報の情報量を増すことに有効である。ID生成子310にて個別に出力レベルを設定することにより、IDデータ信号140の内容を変えることができる。。

2値を出力するID生成子310は、プルアップ／プルダウン抵抗、スイッチ、ジャンパーピン、フリップフロップ、SRAMやDRAM等の揮発メモリ、EEPROMやフラッシュメモリ等の不揮発メモリを用いることができる。また、多値を出力するID生成子310は、抵抗分圧回路、オペアンプなどを用いる。

【0021】

出力活性部200は、ID生成部300の各ID生成子から出力されるIDデータ信号140-1～mを入力とする複数のゲート素子210を有して構成される。ゲート素子210はモジュール選択有効信号130が有効状態の時のみゲートが開かれる素子である。これにより、出力活性部200からは、モジュール選択有効信号130が有効状態のときに、IDデータ信号140がID出力信号150としてモジュールIDバス400に出力される。なお、モジュール選択有効信号130がネゲートされている場合には、ゲート素子210はハイ・インピーダンス状態となりID出力信号150は出力されない。ゲート素子210は、通常の3状態バッファや、MOS（Metal Oxide Semiconductor）のトランスファークゲートを用いることができる。トランスファークゲートを用いる場合は、任意の電圧レベル（素子にて定められる範囲）を、入力から出力へ伝達することが可能となる。

【0022】

ここで、演算処理モジュールによりI/OモジュールのIDデータ信号140を取得する際のタイミングチャートを示す。同図は、演算処理モジュール1と2つのI/Oモジュール2-1、2-2の信号状態を図示しており、縦軸に信号の種類、横軸に時間を示す。まず、モジュール選択信号出力部10が信号線110-

1を駆動するモジュール選択信号30-1をHighレベルにする。これにより、1番目に接続されたI/Oモジュール2-1のモジュール選択有効信号130がアサートされ、これにより出力活性部200のゲート素子210が開き、ID生成部300から0x01（0xは16進数を表すプレフィックス）が出力され、これがID出力信号150としてモジュールIDバス400に出力される。このとき、他のI/Oモジュール2からモジュールIDバス400にID出力信号150が出力されない。したがって、演算処理モジュール1のモジュールID入力部20は、1番目のI/Oモジュール2-1のID出力信号150がモジュールIDバス400を介して取得する。これにより、演算処理モジュール1は、1番目のI/Oモジュール2-1にIDとして0x01を有するI/Oモジュールが接続されていることを認識する。次に、モジュール選択信号出力部10は、2番目のI/Oモジュール2-2のIDを取得するために、モジュール選択信号30-2を駆動し、同様にモジュールID入力部20により2番目のI/Oモジュール2-2のIDである0x02を取得してI/Oモジュール2-2の種類を認識する。

【0023】

以上の動作を、モジュール選択信号30の駆動信号を変更しながら繰り返すことにより、演算処理モジュール1は、自身が管理すべき全てのI/Oモジュール2のID情報を取得することができる。このように、本実施の形態によれば、I/Oモジュール2を複数連結した場合でも、演算処理モジュール1のモジュール選択信号出力部10は、任意の位置にあるI/Oモジュール2のモジュール選択有効信号130を駆動することができる。また、同一回路構成のI/Oモジュール2を複数連結した場合であっても、任意の位置にあるI/Oモジュール2のモジュール選択有効信号130のみを駆動して、選択されたI/Oモジュール2の識別情報を取得できる。

【0024】

したがって、演算処理モジュール1は、計算機システムが組み立てられたとき、設定処理部60からモジュール選択信号出力部10に、接続されているI/Oモジュール2のID情報をモジュールIDバス400へ出力するように指令を出

すと、回路装置 510 を有する I/O モジュール 2 の ID がモジュール ID バス 400 を介してモジュール ID 入力部 20 に入力される。モジュール ID 入力部 20 はモジュール ID バスから ID を取得すると、設定処理部 60 へ ID 情報を通知する。設定処理部 60 は、ID 情報をキーとして設定パラメータテーブル 50 を検索し、対回路装置 510 が接続されたシステムバス 500 用のパラメータを取得する。設定処理部 60 は取得したパラメータを通信制御部 70 へ設定し、回路装置 510 へアクセスするために初期化する。なお、通信制御部 70 はシステムバス信号 80 とシステムバス 500 とを制御するシーケンサを有している。また、設定処理部 60 は取得した ID に基づき、アクセス先の回路装置 510 に対応したデバイスドライバを用意する。デバイスドライバは、回路装置 510 に対して必要な初期設定をおこなう。このようにして、演算処理モジュール 1 は、回路装置 510 に応じてシステムバス 500 のパラメータを変更し、回路装置 510 へアクセスすることができる。また、回路装置 510 を初期化することができる。

【0025】

ここで、設定パラメータテーブル 50 の構成を図 6 に示す。図示のように、設定パラメータテーブル 50 は、ID と、バスのパラメータと、ソフトウェア名とを有して構成される。バスのパラメータは、例えば、バスのプロトコル、アドレス・データ・制御信号のセットアップ/ホールド時間などのタイミング、バス幅などがある。同図の例では、ID 51、バスプロトコル 52、バスのビット幅 53、チップセレクト (CS) に対するアドレス信号のセットアップ時間 54 およびデバイスドライバ 55 を、列方向の要素として示している。また、行方向の要素は各 ID の値に対応したそれぞれのパラメータのセットを示している。例えば、ID の値が 0 x 02 の I/O モジュール 2 は、アクセスするためのバスプロトコルは PC カードバスに準拠し、バスのビット幅は 16 ビット、アドレスセットアップ時間は 6 ns、対応するデバイスドライバは Card.oであることを示している。なお、同図では行、列方向の要素を一部省略して示したが、実際の設定パラメータテーブル 50 では、バスアクセスに必要な全てのパラメータをテーブル化する。また、設定パラメータテーブル 50 は、メモリ上に構築するのが好

適である。例えば、C言語で実装するには、各パラメータをメンバとする構造体を定義し、この構造体の配列を用意し、配列のインデックスとしてIDを用いると検索が容易となる。実際に計算機システムを構築する際、多くのID、すなわち多くのI/Oモジュール2をサポートすることは、保持するデバイスドライバの量や制御の複雑さが増大する点で現実的でない。そのため、実際には対応するIDが限られることになるが、配列を使う方法では対応しないIDが判るように実装することが必要であり、たとえば構造体の特定のメンバにマークをする必要がある。このようにして、IDをキーとして対応するバスパラメータやデバイスドライバを検索することが可能となる。

【0026】

次に、通信制御部70の実施例の構成を図7に示す。通信制御部70は、演算処理モジュール1がシステムバス信号80へのアクセスする場合に、システムバス信号80を制御する手順やタイミングを規定する。具体的には、バスアクセスのためのプロトコル（SRAMインターフェースやDRAMインターフェース、PCカードインターフェースなど）や、バス制御信号間のセットアップ/ホールド時間を制御する。設定処理部60は、パラメータ設定信号61を介してバスアクセスに関するパラメータをパラメータレジスタ71に設定する。パラメータレジスタ71は、必要に応じてバスシーケンサ72-A～72-X（まとめて72とする）をシーケンサ切替信号73により切り替える。例えば、SRAMアクセスのためのバスプロトコルと、DRAMアクセスのためのバスプロトコルとは、アドレスやデータ信号、制御信号の扱いが大きく異なる。このような場合には、バスプロトコル毎にバスシーケンサを切り替えるほうが都合がよい。バスシーケンサ72は、パラメータレジスタ71からの指示に応じて、バスアクセス動作のタイミングを調節する。例えば、バスシーケンサ72は、回路装置510の要求する「チップセレクト信号に対するアドレスセットアップ時間規定」に応じて、アドレスを出力するタイミングを前後させる。

【0027】

次に、設定処理部60の動作を表すフローチャートを図8に示す。設定処理部60は、演算処理モジュール1が管理するI/Oモジュール2を特定し、必要な

バスパラメータを設定し、デバイスである回路装置 510 の初期化を行う。これらの処理は、プロセッサとメモリにより実行されるのが適当である。その際プロセッサは、メモリに格納される命令列に従いこれらの処理を実行する。

【0028】

ここで、図 8 に示すバスの設定処理について説明する。設定処理部 60 は、I/O モジュール 2 におけるモジュール選択有効信号 130 を有効とするようにモジュール選択信号出力部 10 に対して通知する（処理 600）。次に、設定処理部 60 は、モジュール ID バス 400 上の ID を取得するよう、モジュール ID 入力部 20 へ指示を出す（処理 601）。その結果取得された ID を、設定処理部 60 はモジュール ID 入力部 20 から得る。設定処理部 60 は、得られた ID が有効な ID か否かを判定する（処理 602）。モジュール選択信号出力部 10 が選択した I/O モジュールが実在しない場合には、どの I/O モジュールもモジュール ID バス 400 を駆動しない状態、すなわちモジュール ID バス 400 上には無効な ID が存在する状態となる。そのため、設定処理部 60 は得られた ID が有効か判定する必要がある。得られた ID が無効な ID であった場合（処理 602 での “No” 判定）は、全ての I/O モジュールを検査したとして『バス設定終了』に至り、処理を終了する。得られた ID が有効な ID であった場合には、設定処理部 60 は ID をキーとして設定パラメータテーブル 50 から対応するパラメータを検索する（処理 603）。この処理は、設定パラメータテーブル 50 からバスパラメータと回路装置 510 に対応したデバイスドライバの情報を得るためである。その結果、ID に対応した有効なデータが存在するか検査する（処理 604）。対応するパラメータやデバイスドライバが存在しない場合、I/O モジュールが存在しても初期化すべき手順が不明として、エラー処理をおこなう（処理 607）。エラー処理後は『バス設定異常終了』に至り、処理を終了する。エラー処理の内容として、計算機の外部に音や光、コンソール出力で通知することが管理上有効である。この際に設定処理部 60 では、問題が発生した I/O モジュール 2 の物理的な位置は把握できている。そこで、たとえば n 番目の I/O モジュール 2 で問題が発生したならば、音や光を n 回断続させて通知するのが好適である。そして、バスパラメータが得られた場合、設定処理部 60 は

通信処理部 70 に対してバスの初期化をおこなう (処理 605)。通信処理部 70 は、バスプロトコルやバス幅、バスのアクセスタイミングなどを設定される。引き続き、必要に応じてデバイスドライバにより回路装置 510 の初期化をおこなう (処理 606)。ただし、計算機で稼動するオペレーティングシステム (OS) によっては、デバイスドライバが呼び出される順序が規定されており、この時点での設定が困難な場合もある。その場合、設定処理部 60 は、OS が後に適切なデバイスドライバを呼び出せるよう、OS へ I/O モジュール 2 の設定を登録するにとどめる。処理 606 が終了すると、再び処理 600 へ復帰し、前述の処理を継続する。以上の処理を『バス設定終了』に至るまでおこなうことで、全ての I/O モジュールに関するバスの設定を網羅できる。また、処理 605 と処理 606 に関して、例えば処理 605 でおこなう処理をデバイスドライバの初期化ルーチンに含めて実装することが考えられる。この場合、処理 605 のステップは不要となる。また、設定パラメータテーブル 50 に記載するのは ID とデバイスドライバのみでよい。処理 605 のバスパラメータの設定をデバイスドライバの処理に含めるか否かについては、利用する OS にあった適切な手段を選べばよい。

(第 2 実施の形態)

図 9 に、本発明のモジュール排他選択部の他の実施例を示す。本実施例のモジュール排他選択部 100 は、D フリップフロップ 101 を用いて構成したことにある。図 2 の実施例の場合、計算機システムに接続する I/O モジュール 2 の最大台数に応じて、信号線 110、120 の本数を用意しなければならない。例えば、信号線 110、120 が n 本であれば、最大接続できる I/O モジュール 2 は n 台となるから、実装面積の増大やコストの増大が問題となる。

【0029】

この点、図 9 の実施例によれば、最大接続できる I/O モジュール 2 の台数を制限せず、かつ入出力する信号線数を一定とすることができる。すなわち、本実施例のモジュール選択信号は、モジュール選択クロックの信号線 110-c_k とモジュール選択イネーブルの信号線 110-e_n の 2 つの信号からなる。一方のモジュール選択クロックの信号線 110-c_k は、D フリップフロップ 101 の

クロック入力端子CLKと、モジュール選択クロック出力信号の信号線120-c kとに接続される。他方のモジュール選択イネーブルの信号線110-e nは、Dフリップフロップ101のデータ入力端子Dに接続される。Dフリップフロップ101のデータ出力端子Qは、モジュール選択イネーブル出力信号の信号線120-e nと、モジュール選択有効信号130とに接続される。モジュール選択クロック出力信号の信号線120-c kとモジュール選択イネーブル出力信号の信号線120-e nは、隣接するI/Oモジュール2のモジュール選択クロックの信号線110-c kとモジュール選択イネーブルの信号線110-e nとへそれぞれ接続される。Dフリップフロップ101は、クロック入力端子CLKに入力される信号が、LowレベルからHighレベルへ遷移する（立ち上がる）際に、データ入力端子Dの値を取り込みデータ出力端子Qへ出力する。データ出力端子Qの値は、クロック入力端子CLKで信号レベルの立ち上がり変化がない限り保持される。

【0030】

本実施例のモジュール選択動作に関するタイミングチャートを図10に示す。図5と同様に、演算処理モジュール1と、2つのI/Oモジュール2との信号状態を図示する。縦軸に信号の種類、横軸に時間を示す。演算処理モジュール1のモジュール選択信号出力部10は、モジュール選択イネーブル信号30-e nをHighにした後、モジュール選択クロック信号30-c kをLowレベルからHighレベルに遷移させる。その結果、1番目のI/Oモジュール2-1において、モジュール排他選択部100のDフリップフロップにモジュール選択イネーブルの信号線110-e nのレベルが取り込まれ、モジュール選択イネーブル出力信号の信号線120-e nと、モジュール選択有効信号130がアサートされる。1番目のI/Oモジュール2-1におけるIDデータ信号140には、1番目のI/Oモジュール2-1の種類に対応するIDである0x01が出力されている。結局、1番目のI/Oモジュール2-1のID出力信号150には0x01が出力される。この時、モジュールIDバス400には、1番目のI/Oモジュール2-1のみが値を出力する。演算処理モジュール1のモジュールID入力部20は、モジュールIDバス400の値をモジュールID信号40より取得する。

【0031】

このようにして、演算処理モジュール1は、1番目のI/Oモジュール2-1にはIDとして0x01を有するI/Oモジュールが接続されていると識別し、引き続き、モジュール選択信号出力部10は、モジュール選択イネーブル信号30-enをLowレベルに駆動する。さらに、モジュール選択信号出力部10は、モジュール選択クロック信号30-ckをLowレベルに戻した後、再びHighレベルに駆動する。その結果、1番目のI/Oモジュール2-1から出力されたモジュール選択イネーブル出力信号の信号線120-enのレベルが、2番目のI/Oモジュール2-2におけるモジュール排他選択部100のDフリップフロップ101に取り込まれる。その結果、1番目のI/Oモジュールにおけるモジュール選択有効信号130はネゲートされ、2番目のI/Oモジュールにおけるモジュール選択有効信号130はアサートされる。

【0032】

2番目のI/Oモジュール2-2におけるIDデータ信号140には、2番目のI/Oモジュール2-2の種類に対応するIDである0x02が出力されている。結局、2番目のI/Oモジュール2-1のID出力信号150には0x02が出力される。この時、モジュールIDバス400には、2番目のI/Oモジュール2-2のみが値を出力する。演算処理モジュール1のモジュールID入力部20は、モジュールIDバス400の値をモジュールID信号40より取得する。以下同様にして、演算処理モジュール1は、2番目のI/OモジュールにIDとして0x02を有するI/Oモジュールが接続されていることを認識する。同様にしてモジュール選択信号出力部10がモジュール選択クロック信号30-ckをトグル動作（LowレベルからHighレベル、さらにその逆に信号線を駆動）させることにより、モジュールID入力部20は、接続されているI/OモジュールのIDを次々に取得することが可能となる。

【0033】

最終的に、演算処理モジュール1から最遠端のI/Oモジュール2がIDを出力した後は、モジュールIDバスにはIDが出力されない。モジュールID入力部20は、その状態をモジュールIDバス400から判定し、I/Oモジュール

2 の検査を終了する。

(第 3 実施の形態)

図 10 に、モジュール排他選択部の他の実施例を示す。本実施例の特徴は、モジュール排他選択部 100 が、モジュール選択信号出力部 10 と通信してモジュール選択制御をすることにある。モジュール排他選択部 100 は、命令送受信部 102 と、モジュール選択信号スイッチ部 103、制御実行部 104 を有する。命令送受信部 102 と制御実行部 104 との間は、制御要求や要求完了を通知するための要求通知信号 105 で接続されている。制御実行部 104 は、スイッチ動作を指示するためのスイッチ制御信号 106 により、モジュール選択信号スイッチ部 103 と接続する。命令送受信部 102 は、モジュール選択信号出力部 10 からの制御を信号線 110-cmd より受信する。ここで信号線 110-cmd は、単線でも複数線でもよい。単線であれば、信号線のレベル変化によるシリアル通信で、モジュール選択信号出力部 10 と命令送受信部 102 は通信する。複数線であれば、例えば複数のビット幅からなるコマンド信号とデータ信号により通信をおこなう。さらに、モジュール選択信号の信号線 110-cmd とモジュール選択応答信号の信号線 110-ack は個別に用意しなくてもよい。例えば、時分割で排他的に信号線の駆動をおこなえば、同一信号線で共用することが可能である。

【0034】

さて、モジュール選択信号の信号線 110-cmd からの命令を受けた命令送受信部 102 は、受信した命令を解釈し必要な処理を制御実行部 104 に指示する。命令送受信部 102 は、指示した処理の結果を制御実行部 104 より受ける、あるいは自身で処理してモジュール選択応答信号の信号線 110-ack によりモジュール選択信号出力部 10 へ応答する。命令送受信部 102 は、初期状態として設定未完了状態であり、設定完了フラグ 107 はクリア状態である。命令送受信部 102 は所定の処理が完了すると、モジュール選択信号出力部 10 より設定完了命令を受信する。設定完了命令を受けると、命令送受信部 102 は内部で保持する設定完了フラグ 107 をセットし設定完了状態となる。

【0035】

制御実行部 104 は、モジュール選択信号スイッチ部 103 を開閉制御、あるいはモジュール選択有効信号 130 を駆動制御する。制御実行部 104 は、これらの処理を実行するよう命令送受信部 102 より指示を受ける。

【0036】

モジュール選択信号スイッチ部 103 は、制御実行部 104 からのスイッチ開閉制御指示を、スイッチ制御信号 106 を介して受けると、モジュール選択入力信号の信号線 110 とモジュール選択出力信号の信号線の信号線 120 とを接続あるいは切断、すなわちスイッチの開閉制御をおこなう。スイッチ素子として、MOS のトランスファークロッシングのような半導体スイッチや、リレーのような機械スイッチが挙げられる。モジュール選択信号スイッチ部 103 のスイッチの状態として、電源投入直後は開（切断）状態であるとする。なお、上記ではモジュール選択有効信号 130 の駆動制御を制御実行部 104 が指示を受けておこなうとしたが、命令送受信部 102 が直接モジュール選択有効信号 130 を駆動したり、スイッチ開閉制御をしたりしてもよい。

【0037】

図 12 に、本実施の形態のモジュール選択信号出力部 10 から発行される命令パケットの構造を示す。命令パケットは、あて先アドレス 430 とコマンド 431 から構成される。図 12 は、コマンドの一覧を示している。あて先アドレス 430 として、命令送受信部 102 が設定未完了状態の場合に応答を受ける特別なアドレスと、それ以外のアドレスがある。設定完了フラグ 107 がクリア状態の場合、命令送受信部 102 は設定未完了状態であり特別なアドレス（図 12 では 0x00）に応答する。命令送受信部 102 は、全ての命令送受信部 102 の間で独立となるような固有アドレス 108 を有する。命令パケットのあて先アドレス 430 に固有アドレス 108 を指定することで、特定の固有アドレス 108 を有する I/O モジュール 2 に対し、コマンド 431 を送付することが可能となる。命令送受信部 102 が受信するコマンドとして、例えば、命令送受信部 102 の固有アドレス 108 を通知させる Identify コマンド、モジュール選択有効信号 130 を制御するコマンド、モジュール選択信号スイッチ部 103 を制御するコマンド、設定完了フラグ 107 をセットするコマンドがある。

【0038】

図12では、あて先アドレス430として8ビットの数値を挙げたが、それに制限されるものではない。また命令パケットは、モジュール選択信号30の本数に限定されず、単線によるシリアル通信、あるいは複数線による他ビット幅通信のいずれで対応する。

【0039】

図13は、本実施の形態のモジュール選択動作に関するシーケンス図を示す。本シーケンス図は、各構成要素が時系列にそって相互に作用する様子を表すものであり、上から下へ時間が経過することを表す。図13を用いて、複数接続されたI/Oモジュール2のID情報を取得するアルゴリズムを説明する。まず、全てのモジュール選択信号スイッチ部103は開（切断）状態にあるとする。また、全ての命令送受信部102における設定完了フラグ107はクリア状態である。

〔処理1〕モジュール選択信号出力部10は、直接に接続されているI/Oモジュール2を識別するために、Identify命令を送信する（手順451）。Identify命令は、あて先アドレス430として0x00、すなわち設定未完了状態のモジュールを対象とし、かつコマンド431にはIdentify要求を含む命令パケットで表される。はじめは、モジュール選択信号スイッチ103が開状態であるため、当該命令パケットは、演算処理モジュール1に隣接するI/Oモジュール2-1にのみ到達する。I/Oモジュール2-1における命令送受信部102は、受信した命令をデコードする（手順452）。Identify命令を受信した命令送受信部102は、自身が有する固有アドレスを応答する（手順453）。その結果、モジュール選択信号出力部10は、隣接しているI/Oモジュール2-1が有する固有アドレス108を取得する。

〔処理2〕次に、モジュール選択信号出力部10は、モジュール選択命令を送信する（手順454）。モジュール選択命令は、直前に取得した固有アドレス108をあて先アドレス430とし、モジュール選択有効要求を示す0x02をコマンド431に含む命令パケットである。命令送受信部102は、命令をデコードする（手順452）。その結果、命令送受信部102は自身に対するモジュール

選択命令であると解釈する。そこでモジュール選択有効信号 130 をアサートするように、モジュール選択有効要求を制御実行部 104 へ指示する（手順 455）。制御実行部 104 は、モジュール選択有効信号 130 をアサートし、その結果を命令送受信部 102 に返答する（手順 456）。応答を得た命令送受信部 102 は、処理が終了した旨をモジュール選択信号出力部 10 へと返答する（手順 457）。

〔処理 3〕その後、モジュール選択信号出力部 10 はモジュール ID 入力部 20 に通知し、モジュール ID 入力部 20 は ID 情報を取得する。

〔処理 4〕モジュール選択信号出力部 10 は、モジュール選択を解除する命令（コマンド 431 に「モジュール選択無効要求」を含める）を送信する。処理 2 と同様に、当該命令を受信した命令送受信部 102 は、モジュール選択有効信号 130 をネゲートするよう制御実行部 104 に指示する。制御実行部 104 はモジュール選択有効信号 130 をネゲートし、その結果を命令送受信部 102 に返答する。応答を得た命令送受信部 102 は、処理が終了した旨をモジュール選択信号出力部 10 へと返答する。

〔処理 5〕モジュール選択信号出力部 10 は、設定完了命令（コマンド 431 に設定完了フラグセットを含める）を送信する。命令送受信部 10 は、自身が持つ設定完了フラグ 107 を 1 にセットし、応答をモジュール選択信号出力部 10 へ返答する。

〔処理 6〕モジュール選択信号出力部 10 は、スイッチ閉命令（コマンド 431 に「モジュール選択信号スイッチ閉制御」を含める）を送信する（手順 458）。命令をデコードした命令送受信部 10 は、モジュール選択信号スイッチ部 103 を閉じるよう制御実行部 104 へ指示する（手順 459）。制御実行部 104 は、モジュール選択信号スイッチ部 103 を閉じた旨を命令送受信部 10 へ応答する（手順 460）。そして、命令送受信部 102 はスイッチ閉が完了した旨をモジュール選択信号出力部 10 へ通知する（手順 461）。全ての I/O モジュール 2 に対して、モジュール選択信号出力部 10 は以上の処理 1 から処理 6 を繰り返す。処理 1 により、設定完了フラグが 1 にセットされていない命令送受信部 102 を有する I/O モジュール 2 が反応し、固有アドレスを返答する。そして

モジュール選択信号出力部 10 は、処理 2 から処理 6 を順次実行する。

【0040】

以上を繰り返すことにより、演算処理モジュール 1 に隣接する順番に、I/O モジュール 2 は自身の固有アドレスを報告し、当該 I/O モジュールの ID 情報を出力する。よってモジュール ID 入力部 20 は、順番に ID 情報を取得することが可能となる。最終的にモジュール ID 入力部 20 が、全ての I/O モジュール 2 の ID 情報を取得した時点で、本アルゴリズムは終了する。この時点でモジュール選択信号出力部 10 は、物理的な I/O モジュール 2 の接続順と固有アドレス 108 との対応付けが完了している。また同様に I/O モジュール 2 の接続順と ID 情報との対応付けも完了している。

(第 4 実施の形態)

図 14 に、本発明の計算機システムの実施の形態を示す。本実施の形態の特徴は、モジュール選択信号出力部 10 がモジュール ID クロック信号 31 を出力し、ID 生成部 300 はモジュール ID クロック入力信号 111 を受けて ID 出力信号 150 を出力する点が挙げられる。

【0041】

まず、モジュール選択信号出力部 10 は、先に説明した実施の形態と同様の手順によりモジュール選択有効信号 130 をアサートさせる。その後、モジュール選択信号出力部 10 は、モジュール ID クロック信号 31 をトグルする。その結果、モジュール ID クロック入力信号 111 のトグルに同期して、ID 出力信号 150 へ ID 情報が出力される。本実施の形態によれば、モジュール ID 入力部 20 は、少ない本数（例えば 1 本）のモジュール ID バス 400 から多くのビットを有する ID 情報を取得することが可能となる。

【0042】

図 15 に本実施の形態の ID 生成部 300 の構成を示す。ID 生成子 310、セクタ 320、D フリップフロップ 330 を一単位とし、これらが複数個（図 12 では k 個）連結して構成される。ID 生成子 310-1 ~ k は、図 4 と同様である。また、D フリップフロップ 330-1 ~ k は、図 9 と同様である。セクタ 320-1 ~ k は、モジュール選択有効信号 130 がアサートされている場

合、“1”が付された入力信号を選択して出力し、モジュール選択有効信号130がネゲートされている場合、“0”が付された入力信号を選択して出力する素子である。モジュール選択有効信号130がアサートされている際に、モジュールIDクロック入力信号111をトグルすることにより、ID生成子310-1～kの値が順にIDデータ信号140へ出力される。

【0043】

図16に、本実施の形態のID生成部300の動作を表すタイミングチャートをしめす。はじめにIDデータ信号140は、k番目のID生成子310-kの値を出力している。ID生成部300における各Dフリップフロップ330-1～kは、ID生成子310-1～kの値を出力しているものとする。モジュール選択有効信号130がアサートされると、IDデータ信号140の値は出力活性部200を介してモジュールIDバス400へ出力される（時点480）。次に、モジュール選択信号出力部10がモジュールIDクロック信号31をLowレベルからHighレベルへ駆動すると、各Dフリップフロップ330-1～kはセクタ320-1～kの出力を保持しQ端子へ出力する（時点481）。この時点でセクタ320-1～kは全て“1”が付された入力信号を選択している。結局、ID生成部300では、前段のDフリップフロップの値を次段のDフリップフロップが取り込むシフト動作が行われる。その後、モジュールIDクロック信号31をモジュール選択信号出力部10がトグルすると、それに同期してIDデータ信号140へID生成子310の値が順次出力される（時点482～485）。IDデータ信号140に出力された値は、出力活性部200を介してモジュールIDバス400へ出力される。全てのID生成子310の値を出力した後、モジュール選択有効信号130はネゲートされる。その後、モジュール選択信号出力部10は、モジュールIDクロック信号31をLowレベルからHighレベルへと駆動する（時点486）。この動作により、全てのDフリップフロップ330-1～kはID生成子310-1～kの値で初期化される。このようにして、ID生成部300はIDデータ信号140へID情報を出力する。

【0044】

なお、Dフリップフロップ330-1～kの値を初期化するために、モジュ-

ル選択信号出力部 10 は、モジュール選択をする前にモジュール ID クロックを少なくとも 1 回は Low レベルから High レベルへ駆動する必要がある。

(第 5 実施の形態)

図 17 に、本発明の計算機システムの他の実施の形態の構成を示す。本実施の形態の特徴は、ID 生成部 300 と出力活性部 200 とを兼ね備えた ID 出力部 250 を設けたことである。モジュール選択有効信号 130 が ID 出力部 250 に入力されると、ID 出力部 250 で定められる ID 情報が ID 出力信号 150 を介してモジュール ID バス 400 へ出力される。ID 出力部 250 は、駆動する必要がある信号線のみ、Low あるいは High レベルへ駆動する。モジュール選択有効信号 130 がアサートされている場合にモジュール ID バスの複数信号線を Low レベルに駆動する ID 出力部 250 であれば、モジュール ID バス 400 はプルアップ抵抗によりレベルを保持する必要がある。また、モジュール選択有効信号 130 がアサートされている場合にモジュール ID バスの複数信号線を High レベルに駆動する ID 出力部 250 であれば、モジュール ID バス 400 はプルダウン抵抗によりレベルを保持する必要がある。

図 18 に、本実施の形態の ID 出力部 250 の具体的な構成を示す。図において、ID 出力部 250 はダイオード 260 を有する。ダイオード 260 は、モジュール選択有効信号 130 から ID 出力信号 150 の各信号線に直列に配置される。図 18 ではダイオード 260 を全ての ID 出力信号線 150 に接続しているが、実際には出力すべき ID 情報に応じてダイオード 260 を挿入あるいは削除する。ダイオード 260 を削除する場合には、当該信号線は開放となる。これにより、モジュール選択有効信号 130 の挙動如何にかかわらず、該当する ID 出力信号 150 はハイ・インピーダンス状態となる。

【0045】

ここで、モジュール ID バス 400 はプルダウン抵抗により Low レベルが保持されているとする。モジュール選択有効信号 130 がアサートされると、ダイオード 260 からの出力が High レベルとなる。そのため、ダイオード 260 が直列に接続された ID 出力信号 150 の一部分は High レベルを出力し、ダイオード 260 が接続されていない、開放されている ID 出力信号 150 の他の部分は Low

レベルのままとなる。結果として、モジュール ID バス 400 は、モジュール選択有効信号 130 がアサートされると対応する ID 情報が出力される。

【0046】

図 18 において、ダイオード 260 の代わりにオープンコレクタ出力のトランジスタ（あるいはそれに相当するバッファ）を用いてもよい。その場合、モジュール選択有効信号 130 がアサートされると、トランジスタが挿入された ID 出力信号 150 は Low レベルを出力する。この時モジュール ID バス 400 をプルアップ抵抗によって High レベルに保持しておけば、同様に ID 情報をモジュール ID バス 400 へ出力することが可能である。

（第 6 実施の形態）

図 19 に、本発明を適用した計算機システムのモジュール実装方法の実施形態を示す。本実施形態では、1つの演算処理モジュール 1 と複数の I/O モジュール 2 を積層する構成を表している。演算処理モジュール 1 はプリント基板上に回路を構成し、その片面にコネクタ 90 を実装する。コネクタ 90 は、I/O モジュール 2 の片面にあるコネクタ 91 と結合するよう配置されている。同様に I/O モジュール 2 はプリント基板上に回路を構成し、その片面にコネクタ 91、他方の面にコネクタ 92 を実装する。コネクタ 90 は、演算処理モジュール 1 の入出力信号を I/O モジュール 2 へ接続するための部品であり、その入出力信号にはモジュール選択信号 30、モジュール ID 信号 40、システムバス信号 80 を含む。コネクタ 91 は、演算処理モジュール 1 から、あるいは他の I/O モジュールからの入出力信号を I/O モジュール 2 内部の回路へ接続するための部品である。コネクタ 91 は、モジュール選択入力信号の信号線 110、モジュール ID バス 400、システムバス 500 を含む。コネクタ 92 は、I/O モジュール 2 からの入出力信号を他の I/O モジュールへ接続するための部品であり、モジュール選択出力信号の信号線の信号線 120、モジュール ID バス 400、システムバス 500 を含む。コネクタ 92 とコネクタ 91 とは互いに嵌合するよう構成される。同様に、コネクタ 90 とコネクタ 91 とは互いに嵌合する。

【0047】

このように構成することにより、I/O モジュール 2 は演算処理モジュール 1

とも他の I/O モジュール 2 ととも接続することが可能となる。よって、演算処理モジュール 1 に対して、必要な個数だけ I/O モジュール 2 を積層して接続することができ、柔軟に計算機システムを構成することが可能となる。

(第 7 実施の形態)

図 20 に、本発明を適用した計算機システムのモジュール実装方法の他の実施の形態を示す。本実施の形態は、1 つの演算処理モジュール 1 と複数の I/O モジュール 2 をバックボード 93 上に接続する構成を表している。

【0048】

バックボード 93 は、コネクタ 94 とコネクタ 95-1 ~ 95-n を有し、それらのコネクタはそれぞれ基板を挿入保持するスリットを有し、演算処理モジュール 1 の基板上の接点や I/O モジュール 2 の基板上の接点と、電氣的に接続するための電極を有する。コネクタ 94、95-1 ~ 95-n は、スリットに挿入された基板の接点と、バックボード上の配線とを接続する。コネクタ 94 とコネクタ 95-1 との間は、モジュール選択信号 30 とモジュール選択入力信号の信号線 110、モジュール ID 信号 40 とモジュール ID バス 400、システムバス信号 80 とシステムバス 500 が、それぞれ接続するようバックボード 93 上で配線する。また、コネクタ 95-j とコネクタ 95- {j+1} (ただし j = 1 ~ n-1) との間は、コネクタ 95-j のモジュール選択出力信号の信号線 120 とコネクタ 95- {j+1} のモジュール選択入力信号の信号線 110、コネクタ 95-j のモジュール ID バス 400 とコネクタ 95- {j+1} のモジュール ID バス 400、コネクタ 95-j のシステムバス 500 とコネクタ 95- {j+1} のシステムバス 500 が、それぞれ接続するようバックボード 94 上で配線する。コネクタ 94 に演算処理モジュール 1 を挿入し、95-1 ~ 95-n には演算処理モジュール 1 からの配線が連結するよう I/O モジュール 2 を挿入することで、計算機を構成する。

【0049】

なお、図 20 では基板を直接コネクタのスリットに挿入する構成を示したが、基板上とバックボード 93 の両方にコネクタを実装してもよい。その場合、それぞれのコネクタは勘合するよう構成する。基板上のコネクタをバックボード上の

コネクタに結合することで、計算機を構成する。

【0050】

また、バックボード93と演算処理モジュール1とは同一基板に形成しても良い。その場合、演算処理モジュール1上にコネクタ95を実装し、コネクタ95にI/Oモジュール2を接続する構成となる。このような構成とすることで、バックボードを個別に用意することなく、拡張性を確保することができる。

【0051】

以上説明したように、本発明の各実施の形態によれば、プロセッサがバスに接続される複数のデバイスをバス方式によらずに認識し、任意のバス方式を持つデバイスへアクセスできる。また、モジュール毎に手作業による設定を不要とし、同一回路のモジュールを組み合わせた場合でも、プロセッサはデバイスを認識することが可能となる。

【0052】

【発明の効果】

以上述べたように、本発明によれば、特定のバス方式に制約されることなく、演算処理モジュールに接続されたI/Oモジュールの種類等を識別でき、そのI/Oモジュールのデバイスにアクセス可能にすることができる。

【図面の簡単な説明】

【図1】

本発明に係る計算機システムの第1実施形態の全体構成図である。

【図2】

第1実施形態のモジュール排他選択部の詳細構成図である。

【図3】

第1実施形態のID生成部の詳細構成図である。

【図4】

第1実施形態の出力活性部の詳細構成図である。

【図5】

第1実施の形態の動作を説明するタイミングチャートである。

【図6】

第 1 実施形態の設定パラメータテーブルの構成を示す図である。

【図 7】

第 1 実施形態の通信制御部の構成を示す図である。

【図 8】

第 1 実施形態のバスの初期化の手順を示すフローチャートである。

【図 9】

本発明の計算機システムの第 2 実施形態に係るモジュール排他選択部の詳細構成である。

【図 10】

第 2 実施形態の動作を説明するタイミングチャートである。

【図 11】

本発明の計算機システムの第 3 実施形態に係るモジュール排他選択部の詳細構成である。

【図 12】

第 3 実施形態のモジュール選択信号出力部から発行される命令パケットの構成図である。

【図 13】

第 3 実施形態の処理手順を示すフローチャートである。

【図 14】

本発明の計算機システムの第 4 実施形態に係る全体構成図である。

【図 15】

第 4 実施形態の ID 生成部に係る詳細構成図である。

【図 16】

第 4 実施形態の動作を説明するタイミングチャートである。

【図 17】

本発明の計算機システムの第 5 実施形態に係る全体構成図である。

【図 18】

第 5 実施形態の ID 出力部に係る詳細構成図である。

【図 19】

本発明に係る計算機システムのモジュール実装方法の一実施形態を示す図である。

【図 2 0】

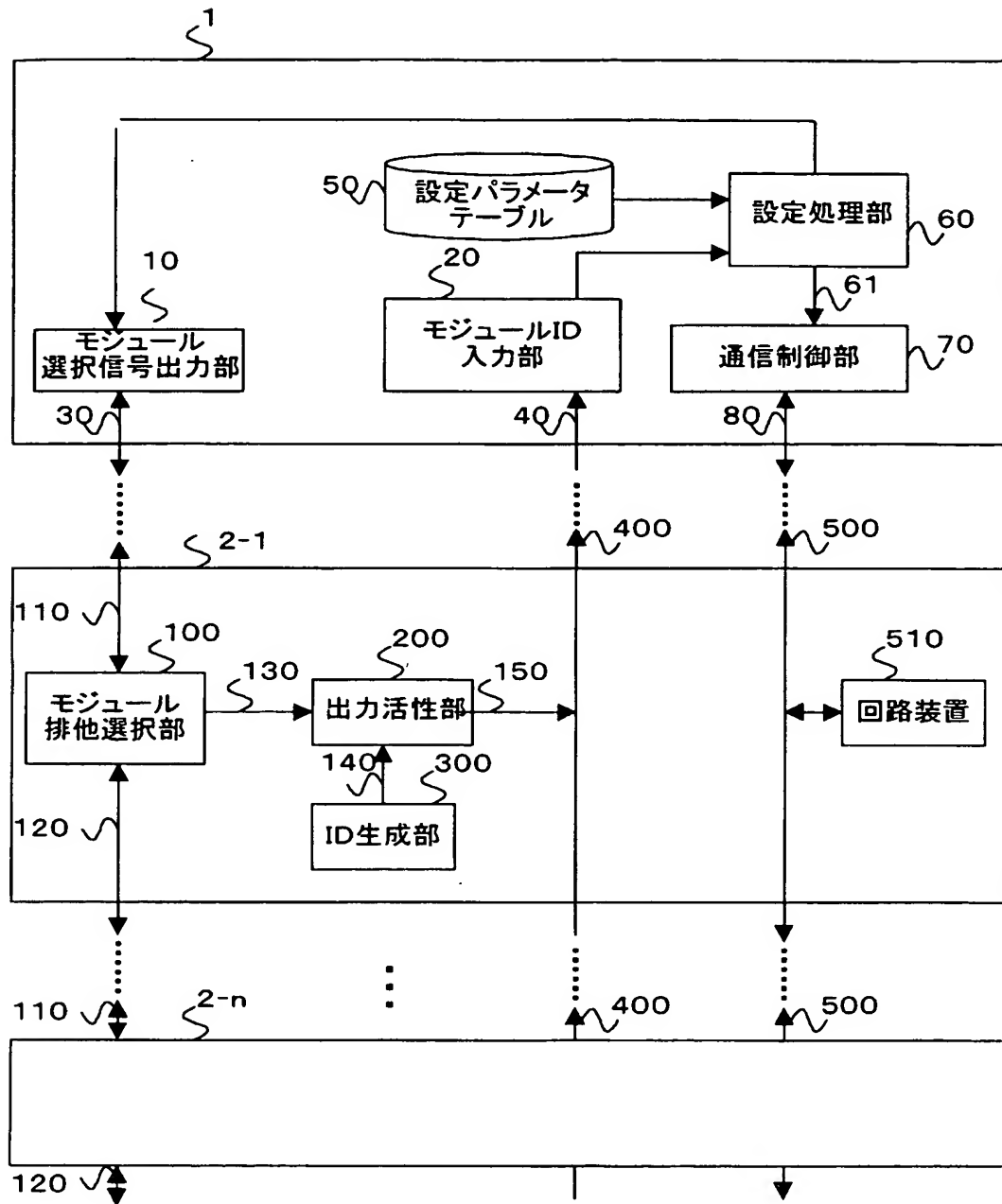
本発明に係る計算機システムのモジュール実装方法の他の実施形態を示す図である。

【符号の説明】

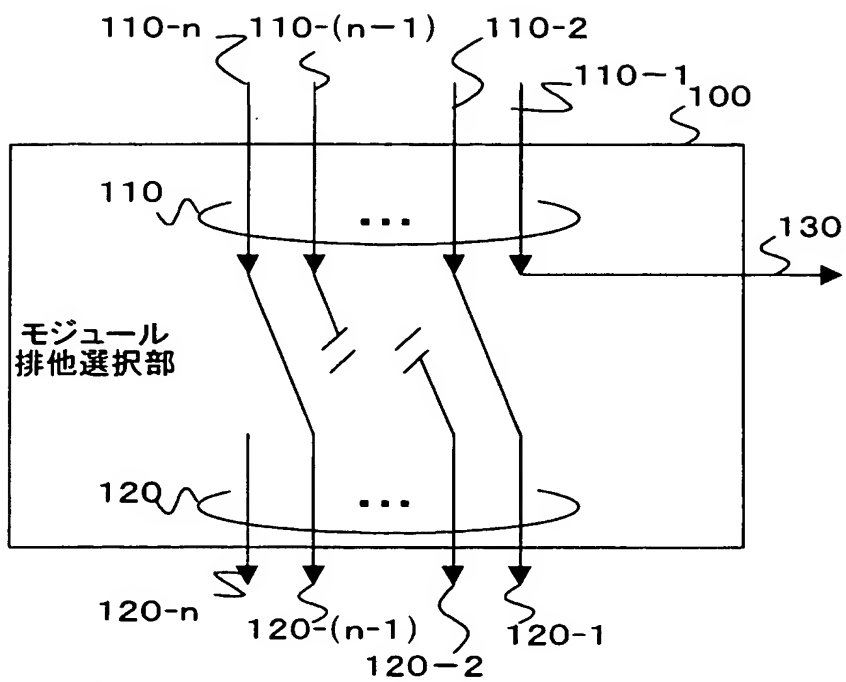
1…演算処理モジュール、2…I/Oモジュール、10…モジュール選択信号出力部、20…モジュールID入力部、30…モジュール選択信号、40…モジュールID信号、50…設定パラメータテーブル、60…設定処理部、70…通信制御部、80…システムバス信号、90～92…コネクタ、93…バックボード、94～95…コネクタ、100…モジュール排他選択部、110…信号線、120…信号線、130…モジュール選択有効信号、140…ID、210…ゲート素子、250…ID出力部、300…ID生成部、400…モジュールIDバス、500…システムバス、510…回路装置

【書類名】 図面

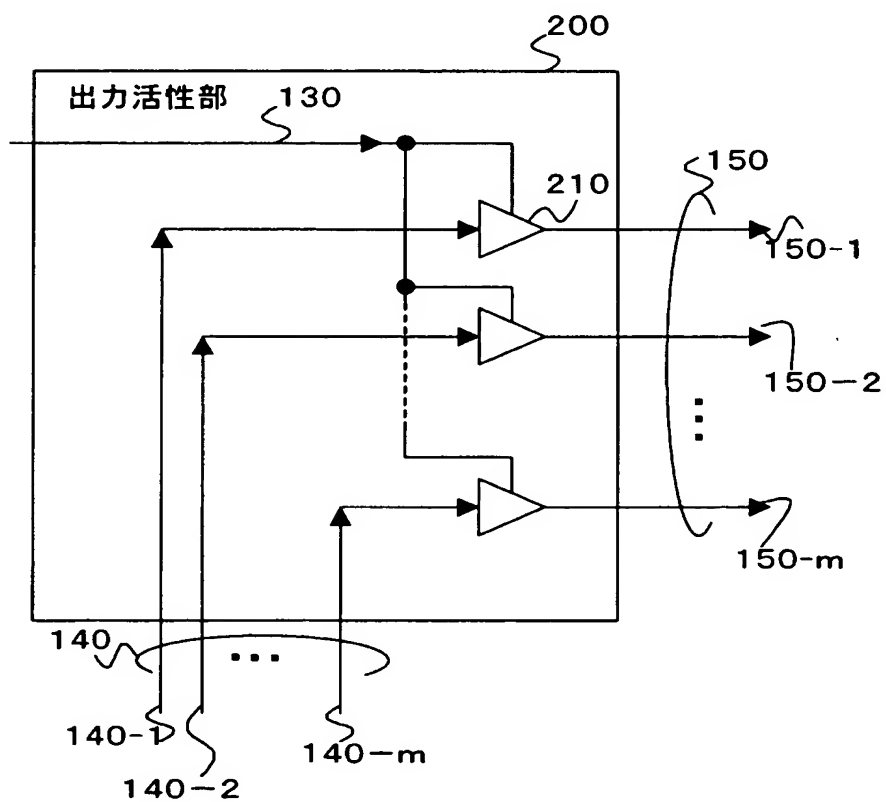
【図 1】



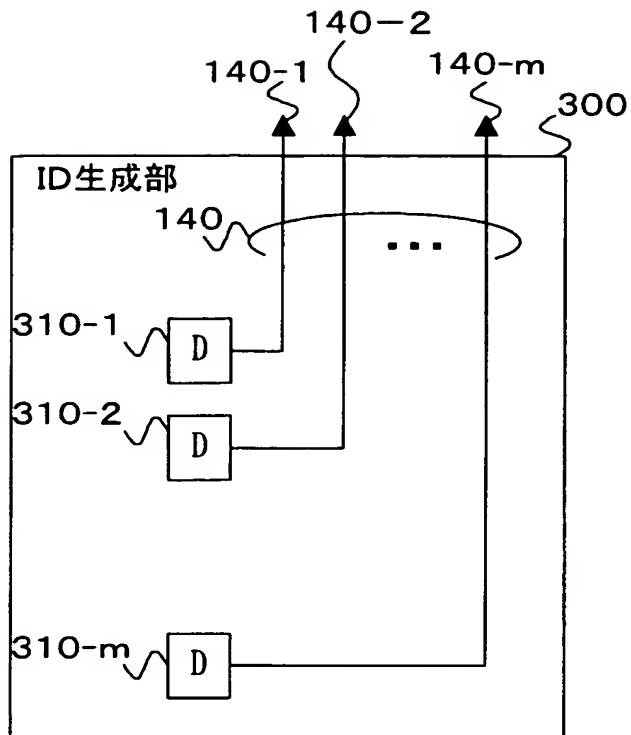
【図 2】



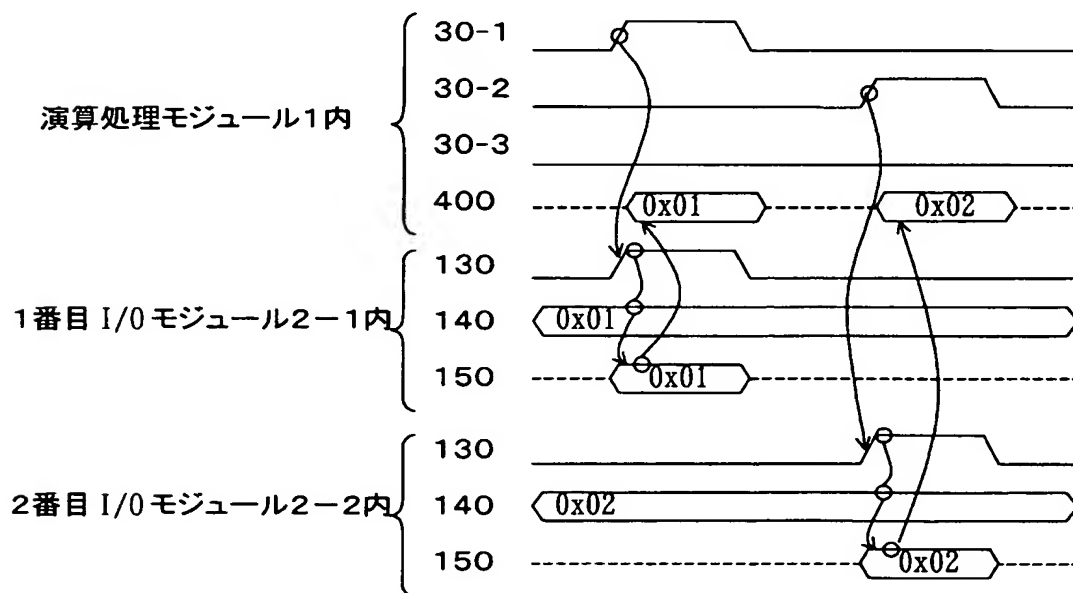
【図 3】



【図4】



【図5】

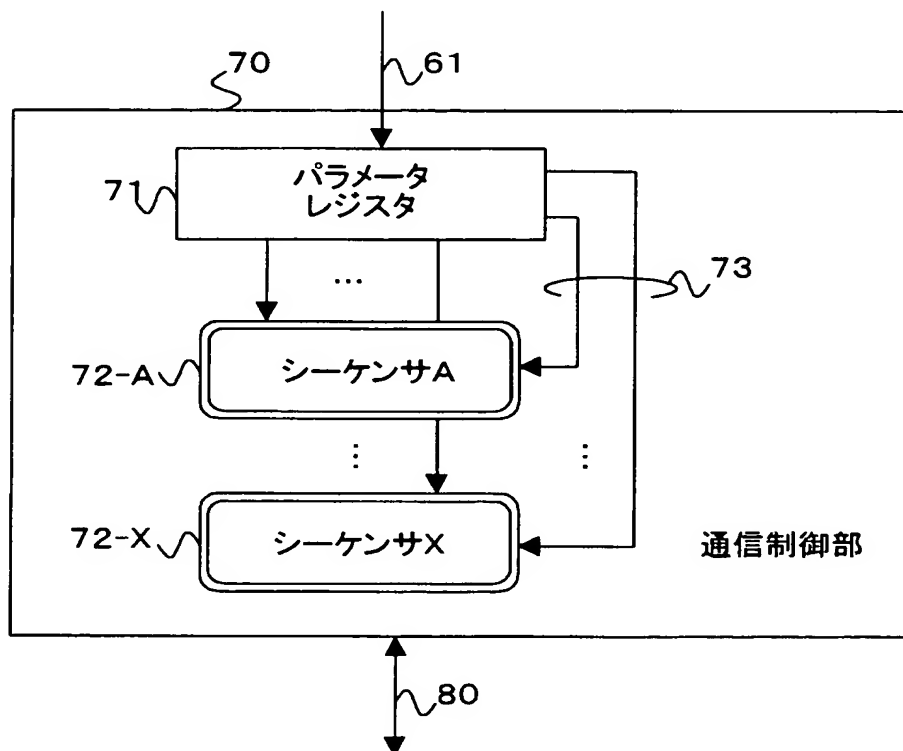


【図 6】

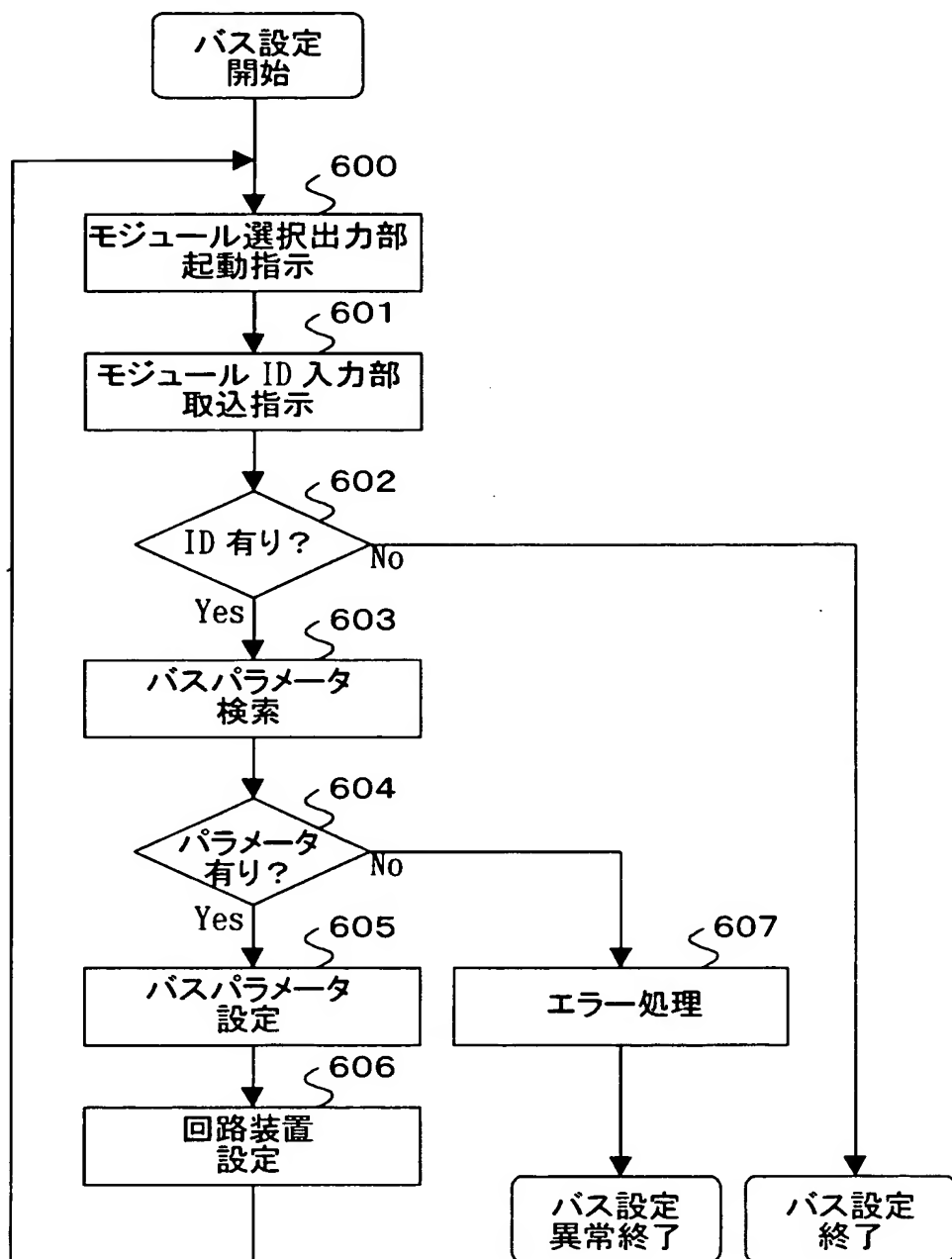
50

51	52	53	54	...	55
ID	バス プロトコル	バス幅	CS に対する アドレスセットアップ時間		デバイス ドライバ
0x01	SRAM	8	5ns		UART.
0x02	PC カード	16	6ns		Card.
⋮					

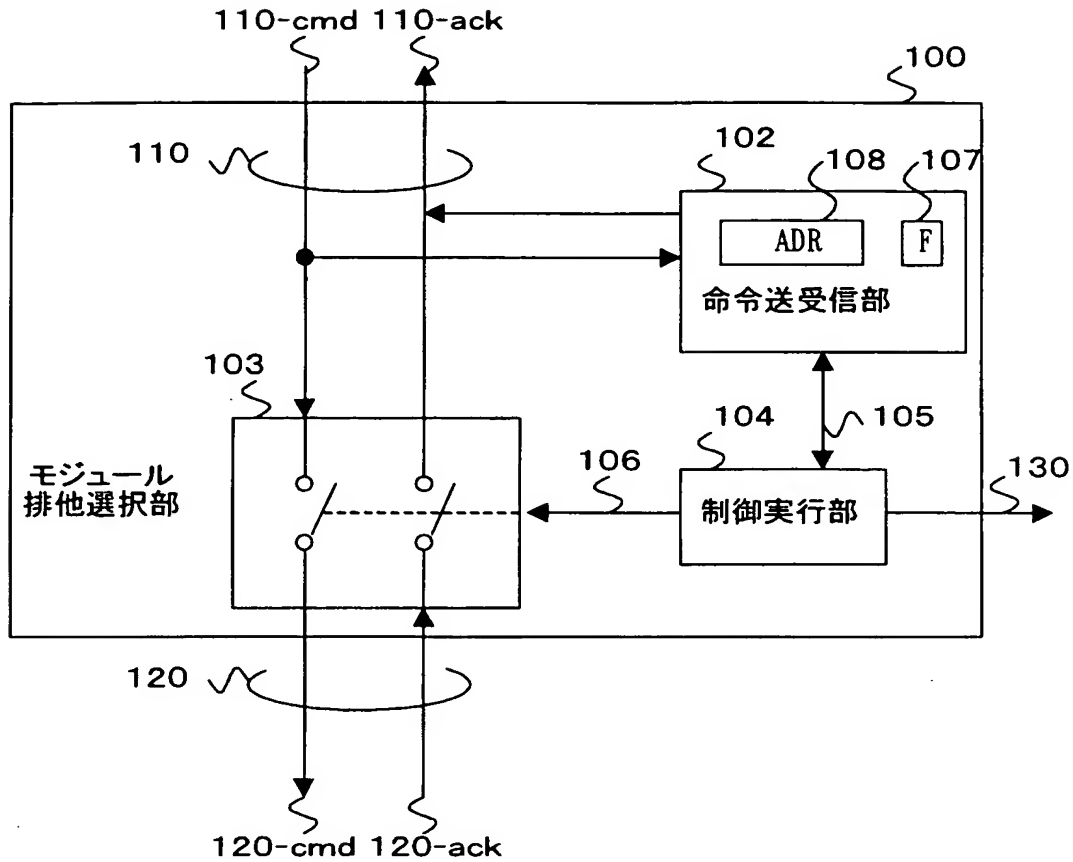
【図 7】



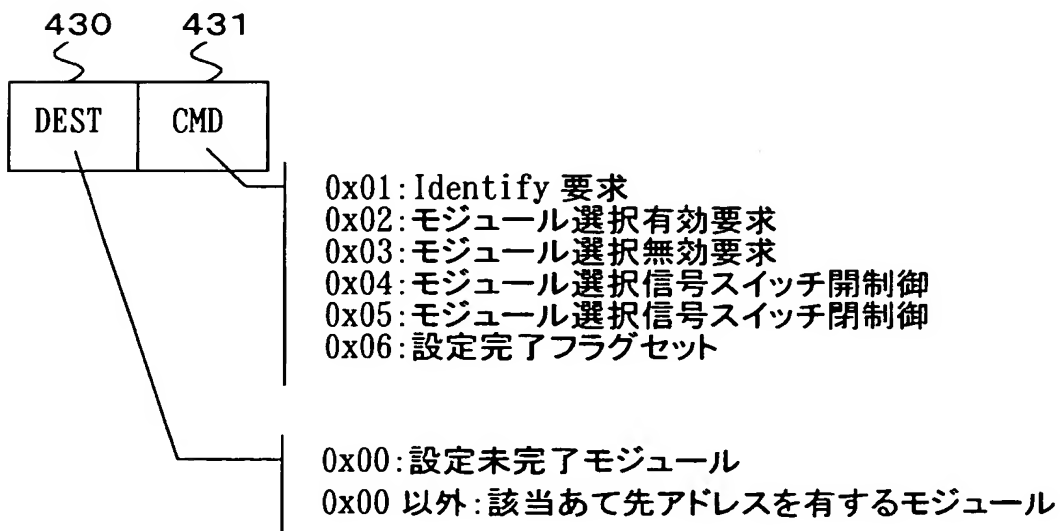
【図 8】



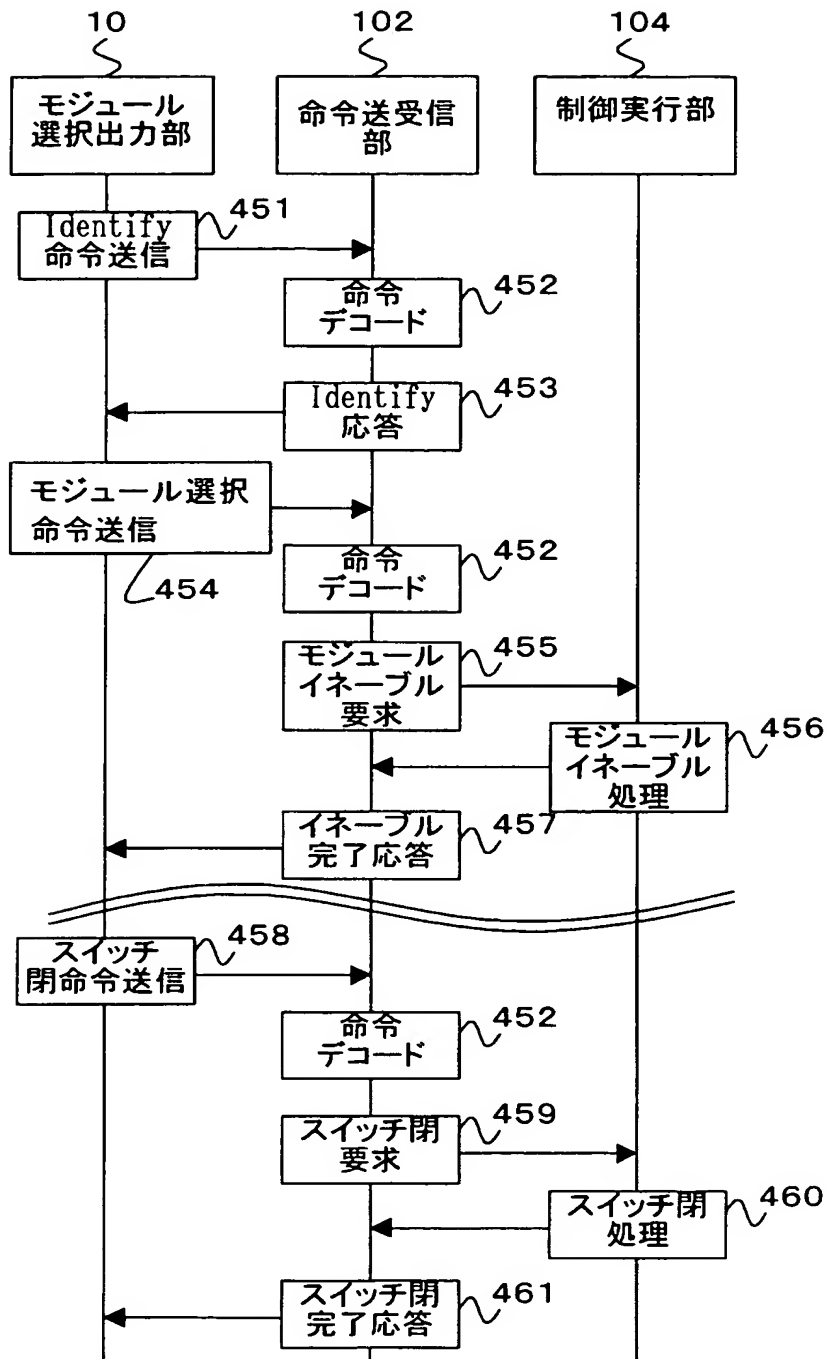
【図 11】



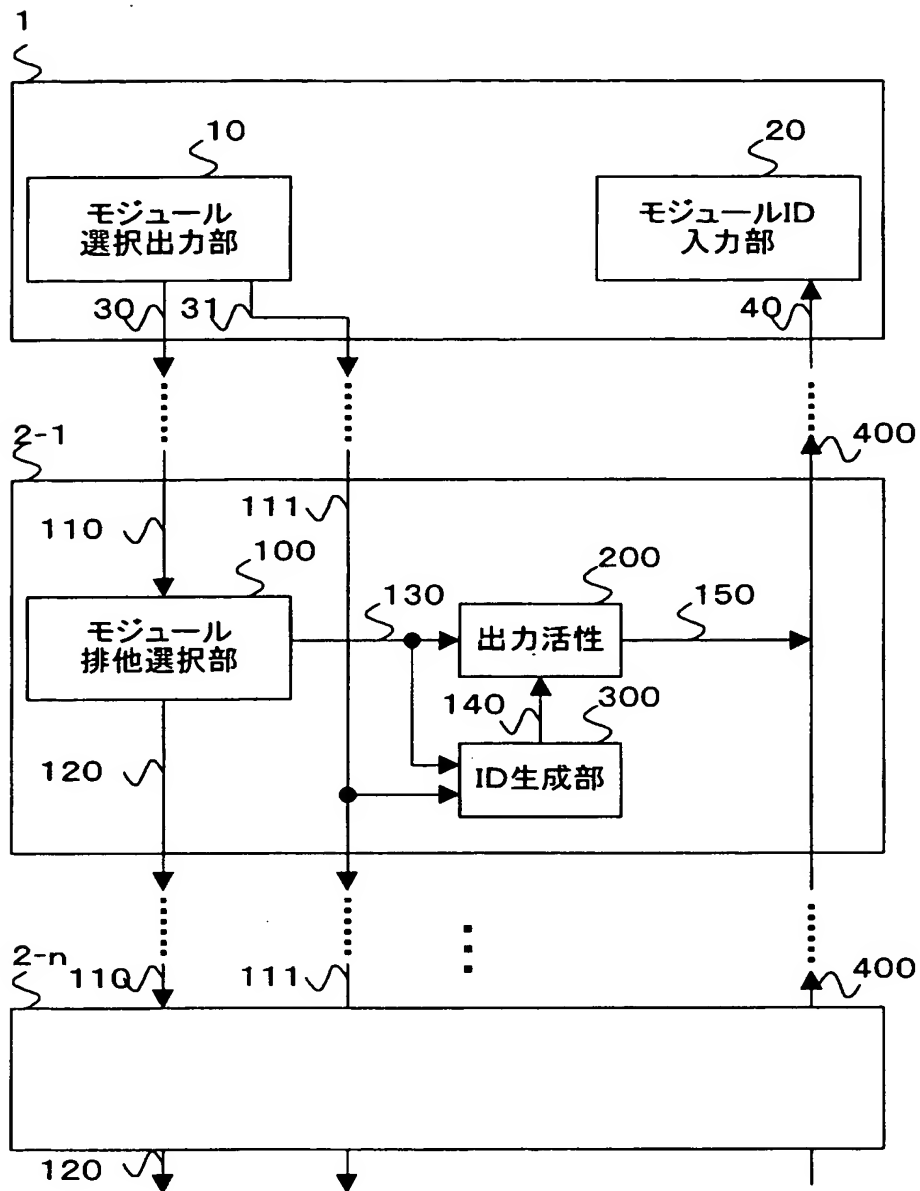
【図 12】



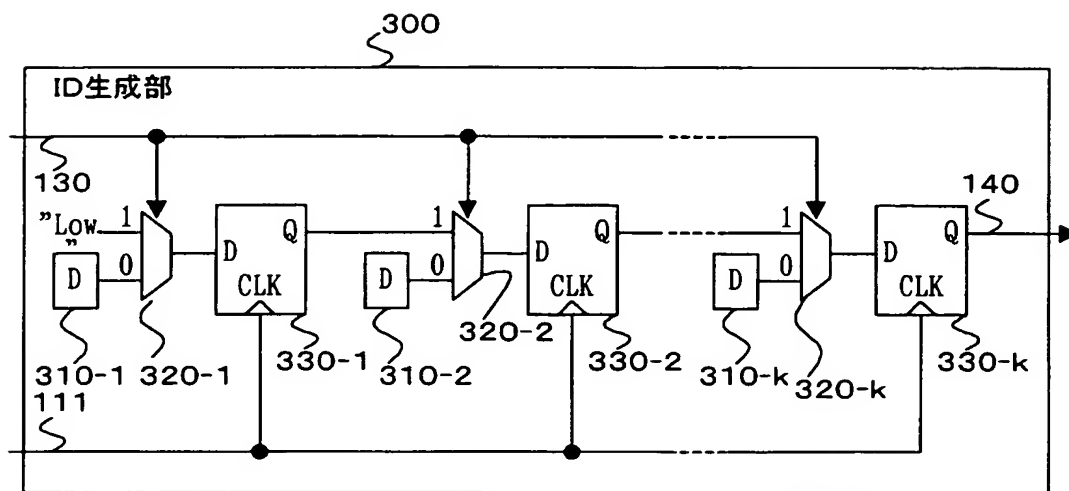
【図 13】



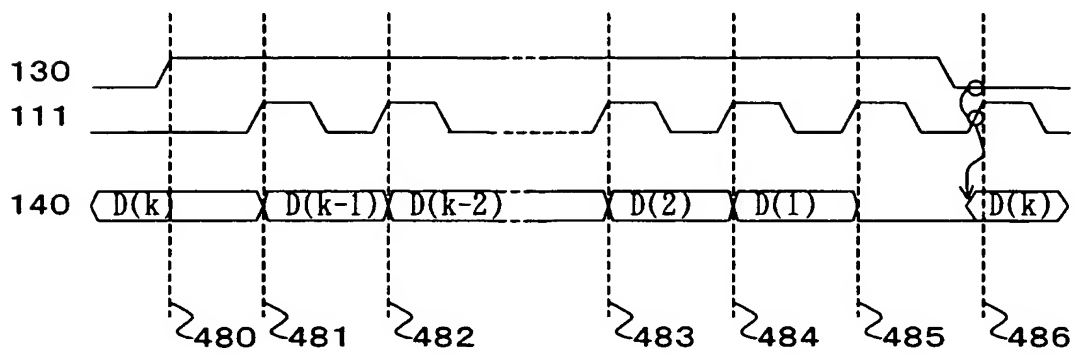
【図14】



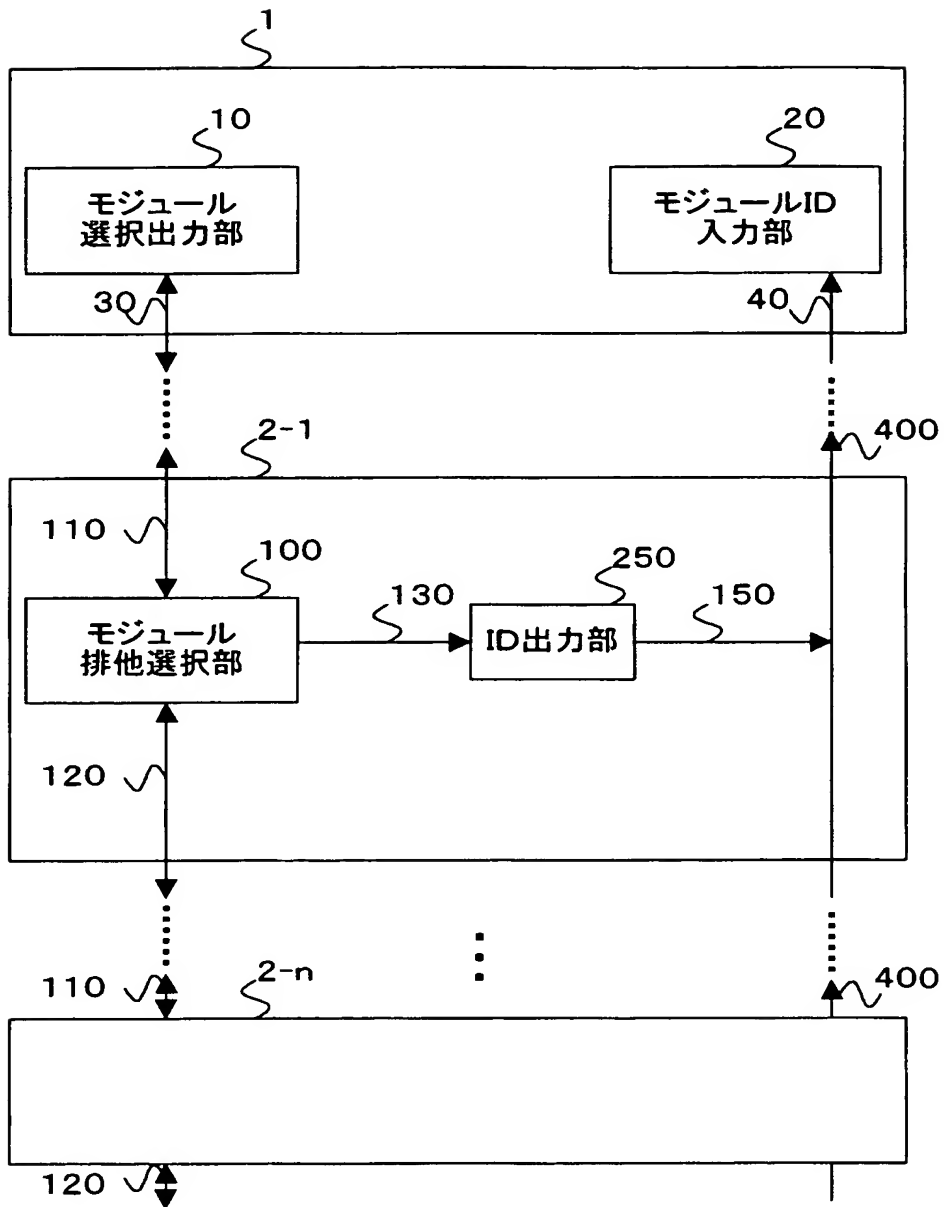
【図 15】



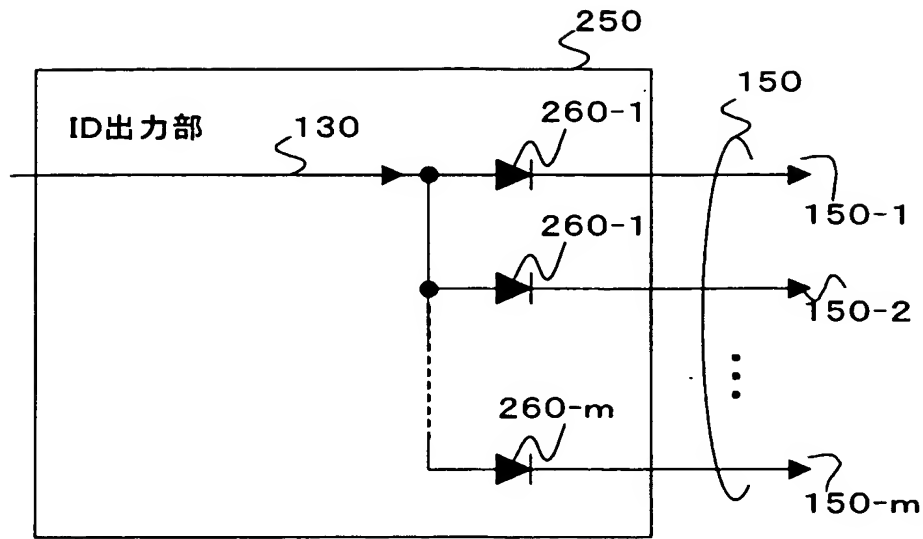
【図 16】



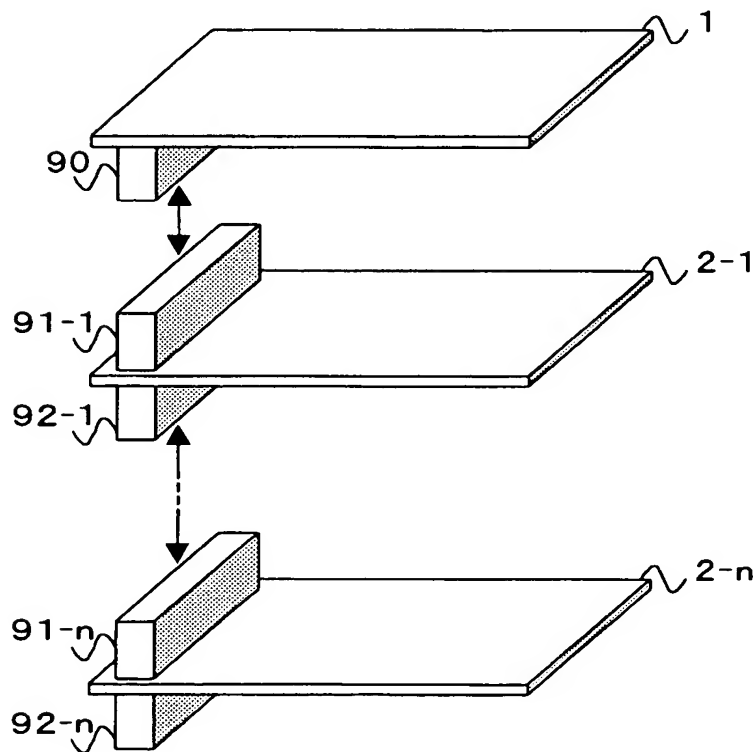
【図 17】



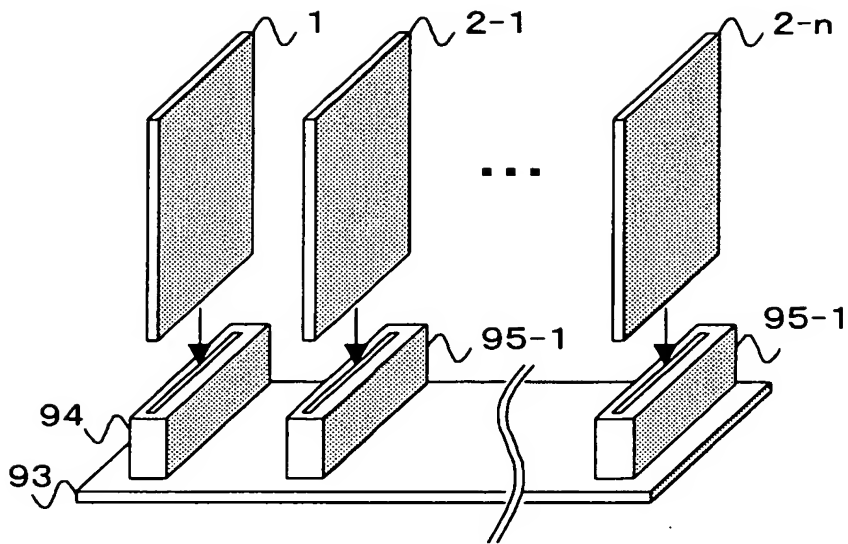
【図 18】



【図 19】



【図 20】



【書類名】 要約書

【要約】

【課題】 特定のバス方式に制約されることなく、演算処理モジュールに接続された I/O モジュールの種類を識別可能にする。

【解決手段】 演算処理モジュール 1 にコネクタを介して層状に接続される各 I/O モジュールにモジュール排他選択部を設け、このモジュール排他選択部により演算処理モジュール側コネクタの同一位置の端子から入力されるモジュール選択信号のみを有効と判別し、これに基づいて自己の I/O モジュールの識別情報をコネクタの所定端子に出力することにより、演算処理モジュールは特定のバス方式に制約されることなく、コネクタの所定端子から I/O モジュールの識別情報を獲得できる。また、演算処理モジュールから順次出力されるモジュール選択信号を、I/O モジュールの接続順に応じて演算処理モジュール側コネクタの同一位置の端子に入力するという簡単なモジュール選択回路方式によって唯一の各 I/O モジュールを選択できる。

【選択図】 図 1

特願 2 0 0 3 - 1 1 8 7 1 4

出 願 人 履 歴 情 報

識別番号

[0 0 0 0 0 5 1 0 8]

1. 変更年月日

1 9 9 0 年 8 月 3 1 日

[変更理由]

新規登録

住 所

東京都千代田区神田駿河台 4 丁目 6 番地

氏 名

株式会社日立製作所